CLAIMS

Sub a1

1. A method of compilation of source program using one or more associated libraries, the method comprising:

identifying one or more instances available for use in the one or more libraries;

receiving a first request to create a first instance during compilation of the source program; and

determining whether the first instance has been identified in the one or more libraries.

2. A method as recited in claim 1, wherein the method further comprises:

creating the first instance when the determining determines that the first instance has not been identified in the one or more libraries and not creating the first instance when the first instance has been identified in the one or more libraries.

09595147 . 101900

3. A method as recited in claim 2, wherein the identifying of one or more instances available for use in the one or more libraries further comprises:

identifying linker symbol names for instances available for use in the one or more libraries, and

wherein the creating of the first instance operates to create the first instance when the linker symbol name for the first instance does not match any of the identified linker symbol names for instances available for use in the one or more libraries.

4. A method as recited in claim 2, wherein the identifying of one or more instances available for use in the one or more libraries further comprises:

accessing the one or more libraries;

examining linker symbol names in symbol tables within the one or more libraries;

selecting linker symbol names that are likely to correspond to instances available for use in the one or more libraries; and

saving the selected linker symbol names.

5. A method as recited in claim 4, wherein the examining of symbol tables is done to extract all linker symbol names that are likely to correspond to instances.

6. A method as recited in claim 4, wherein the selecting of the linker symbol names that are likely to correspond to instances is done by selecting linker symbol names that include a predetermined sequence of characters.

7. A method as recited in claim 4, wherein the saving of the selected linker symbol names is done by using a hash table.

8. A method as recited in claim 4,

wherein determining whether the first instance has been identified in the one or more libraries further comprises:

obtaining a first linker symbol name for the first instance;

comparing the first linker symbol name with those selected linker symbol names that are likely to correspond to template instances, and

wherein creating the first instance operates to create the first instance when the first linker symbol does not match any of those selected linker symbol names that are likely to correspond to template instances.

~~9. A method as recited in claim 1, wherein the source program is written in C++ or Ada programming language.~~

~~10. A compiler system suitable for compilation of source programs, the compiler system comprising:~~

~~a source program;~~

~~a library including at least one instance available for use by the source program;~~

~~an enhanced compiler suitable for compilation of source code, wherein the enhanced compiler accesses the library to identify the one instance available in the library.~~

~~11. A compiler system as recited in claim 10, wherein the enhanced compiler further comprises:~~

~~an instance extractor for extracting the at least one instant available for use by the source program.~~

~~12. A compiler system as recited in claim 11, wherein the enhanced compiler further comprises:~~

~~an instance name comparator operating to compare the at least one instant available with a desired instant.~~

~~13. A compiler system as recited in claim 12, wherein the enhanced compiler further comprises:~~

~~an instance name storage suitable for storage of the at least one instant available for use by the source program.~~

14. A method of compilation of source program using one or more associated libraries with instances available for use by the source program, the method comprising:

examining a linker name table of the one or more associated libraries;

extracting from the linker name table one or more linker symbol names that are likely to correspond to instances;

storing the one or more linker symbol names that have been extracted as one or more stored linker symbol names;

receiving a first request to create a first instance during compilation of the source program, said first instance having a first linker symbol name;

comparing the first linker symbol name with the one or more stored linker symbol names; and

creating the first instance only when said comparing indicates that the first linker symbol name is not one of the stored linker symbol names.

09693517 .101900

15. A method as recited in claim 14, wherein the comparing of the first linker symbol name with the one or more stored linker symbol names is done without transforming the linker symbol names of the one or more libraries.

16. A method as recited in claim 14, wherein the source program is in C+ + or Ada, and wherein storing at least one linker symbol is done by using a hash table.

17. A computer readable media including computer program code for compilation of source program using one or more associated libraries having instances available for use by the source program, the computer readable media comprising

computer program code for identifying one or more instances available for use in the one or more libraries;

Sub a1

computer program code for receiving a first request to create a first instance during compilation of the source program;

computer program code for determining whether the first instance is available for use in the one or more libraries; and

computer program code for creating the first instance when the first instance has not been identified in the one or more libraries and not creating the first instance when the first instance has been identified in the one or more libraries.

18. A computer readable media as recited in claim 17,

wherein the computer program code for identifying of one or more instances available for use in the one or more libraries further comprises:

computer program code for identifying linker symbol names for instances available for use in the one or more libraries, and

wherein the computer program code for creating the first instance operates to create the first instance when the linker symbol name for the first instance does not match any of the identified linker symbol names for instances available for use in the one or more libraries.

19. A computer readable media as recited in claim 17, wherein the computer program code for identifying of one or more instances available for use in the one or more libraries further comprises:

computer program code for accessing the one or more libraries;

computer program code for examining linker symbol names in symbol tables within the one or more libraries;

computer program code for selecting linker symbol names that are likely to correspond to instances available for use in the one or more libraries; and

computer program code for saving the selected linker symbol names.

~~20. A computer readable media as recited in claim 19, wherein the computer program code selecting of the linker symbol names that are likely to correspond to instances is done by selecting linker symbol names that include a predetermined sequence of characters.~~

09659357.101900